# Package: VFS (via r-universe)

August 27, 2024

**Title** Vegetated Filter Strip and Erosion Model

**Version** 1.0.4

**Date** 2018-10-01

**Depends** R (>= 3.4.0)

**Imports** stats, graphics, e1071, nleqslv (>= 3.3.0)

**Suggests** knitr, rmarkdown, testthat

**VignetteBuilder** knitr

**Description** Empirical models for runoff, erosion, and phosphorus loss across a vegetated filter strip, given slope, soils, climate, and vegetation (Gall et al., 2018) <doi:10.1007/s00477-017-1505-x>. It also includes functions for deriving climate parameters from measured daily weather data, and for simulating rainfall. Models implemented include MUSLE (Williams, 1975) and APLE (Vadas et al., 2009 <doi:10.2134/jeq2008.0337>).

**License** GPL-3

**BugReports** https://github.com/sgoslee/VFS/issues

**LazyData** true

**Repository** https://sarahgoslee-usda.r-universe.dev

**RemoteUrl** https://github.com/sarahgoslee-usda/vfs

**RemoteRef** HEAD

**RemoteSha** f2592289308916a3a0d781fe8e4bb63a8e489602

## Contents


VFS-package . . . . . . . . . . 2
APLE . . . . . . . . . . 3
bufferdat . . . . . . . . . . 5
MUSLE . . . . . . . . . . 6
MUSLE.K . . . . . . . . . . 8
MUSLE.LS . . . . . . . . . . 9

peak . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10
print.APLE . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 11
print.VFS . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 12
rainfall . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13
read.dly . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 14
soildat . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 16
summary.APLE . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 17
summary.VFS . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 18
temperature . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20
USC00368449.dly . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 21
VFS . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 22
VFSAPLE . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 25
weather . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 27
wth.param . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 28

**Index** **31**

---

`VFS-package` *Vegetated Filter Strip and Erosion Model*

---

## Description

Empirical models for runoff, erosion, and phosphorus loss across a vegetated filter strip, given slope, soils, climate, and vegetation (Gall et al., 2018) <doi:10.1007/s00477-017-1505-x>. It also includes functions for deriving climate parameters from measured daily weather data, and for simulating rainfall. Models implemented include MUSLE (Williams, 1975) and APLE (Vadas et al., 2009 <doi:10.2134/jeq2008.0337>).

## Details

The DESCRIPTION file: This package was not yet installed at build time.

Index: This package was not yet installed at build time.
This package implements runoff, erosion, filter strip, and phosphorus loss models in R.

## Author(s)

Sarah Goslee [aut, cre], Heather Gall [aut], Tamie Veith [aut]

Maintainer: Sarah Goslee <Sarah.Goslee@ars.usda.gov>

## References

Gall, H. E., Schultz, D., Veith, T. L, Goslee, S. C., Mejia, A., Harman, C. J., Raj, C. and Patterson, P. H. (2018) The effects of disproportional load contributions on quantifying vegetated filter strip sediment trapping efficiencies. *Stoch Environ Res Risk Assess* **32**(8), 2369–2380. doi:10.1007/s004770171505x

### Examples

```
# state college GHCN data
#
# weather <- read.dly(system.file("extdata", "USC00368449.dly", package = "VFS"))
data("weather") # same object

weather.param <- wth.param(weather, method="markov")

rain.compare <- rainfall(365*3, weather.param)
temp.compare <- temperature(365*3, weather.param)

data(soildat)
data(bufferdat)

# bluegrass buffer, clay loam soil
vfs.CL <- VFS(nyears = 3, thissoil = subset(soildat, Soil == "CL"),
rain=rain.compare, temperature=temp.compare,
thisbuffer = subset(bufferdat, Species == "bluegrass"), Duration = 2,
FieldArea = 4000, VFSwidth = 10.7, VFSslope = 0.02,
z = 1000, b =1.5)

print(vfs.CL)
summary(vfs.CL)

aple.CL <- VFSAPLE(vfs.CL, soilP = 120, OM = 2)

print(aple.CL)
summary(aple.CL)
```

---

## APLE *Agricultural Phosphorus Loss Estimator*

---

### Description

Agricultural loss of phosphorus based on soil phosphorus level, additions of fertilizer and manure, and erosion.

### Usage

```
APLE(soilP, clay, OM, precip, runoff, erosion, manureP = 25,
    manureSolids = 25, manureWEP = 50, manureIn = 40,
 fertP = 10, fertIn = 40)
```

### Arguments

| | |
|---|---|
| soilP | soil test Mehlich 3 phosphorus (mg/kg). |
| clay | soil clay (%). |

| | |
|---|---|
| OM | soil organic matter (%). |
| precip | annual precipitation (in). |
| runoff | annual runoff (in) |
| erosion | annual erosion (ton/ac). |
| manureP | manure P applied (kg/ha). |
| manureSolids | manure solids (%). |
| manureWEP | manure water-extractable phosphorus/TP (%). |
| manureIn | manure incorporated (%). |
| fertP | fertilizer phosphorus applied (kg/ha). |
| fertIn | fertilizer incorporated (%). |

## Details

This function implements the basic version of the spreadsheet-based Agricultural Phosphorus Loss Estimator model (APLE) in R, and is vectorized. This model calculates annual phosphorus loss by compartment (due to erosion, dissolved soil phosphorus, dissolved manure, dissolved fertilizer) based rainfall, soil properties, and management. The units match those of the original spreadsheet.

## Value

| | |
|---|---|
| lossErosion | soil erosion phosphorus loss (kg/ha). |
| lossDissolvedSoil | soil dissolved phosphorus loss (kg/ha). |
| lossDissolvedManure | manure dissolved phosphorus loss (kg/ha). |
| lossDissolvedFertilizer | fertilizer dissolved phosphorus loss (kg/ha). |
| lossTotal | total phosphorus loss (kg/ha). |

## Author(s)

Sarah Goslee

## References

Vadas, P. A., Good, L. W., Moore, P. A., Jr. and Widman, N. (2009) Estimating phosphorus loss in runoff from manure and fertilizer for a phosphorus loss quantification tool. *J Environ Qual* **38**, 1645–1653. doi:10.2134/jeq2008.0337

## See Also

VFSAPLE

## Examples

```
APLE(soilP = 127, clay = 17, OM = 6, precip = 35, runoff = 6,
erosion = 7, manureP = 25, manureSolids = 25, manureWEP = 50,
manureIn = 40, fertP = 10, fertIn = 40)
```

---

# bufferdat *Parameters for vegetated buffers*

---

## Description

Contains parameters describing vegetated filter strips for use in VFS modeling.

## Usage

```
data("bufferdat")
```

## Format

A data frame with 2 observations on the following 3 variables.

`Species` Type of buffer

`bg` Average stem spacing (cm)

`n` Manning's roughness coefficient (s m^(-1/3))

## Details

Currently contains data for a cool-season and a warm-season grass buffer.

## Source

Haan CT, Barfield BJ, Hayes JC (1994) Design hydrology and sedimentology for small catchments. Acad. Press, San Diego

## Examples

```
# state college GHCN data
#
# weather <- read.dly(system.file("extdata", "USC00368449.dly", package = "VFS"))
data("weather") # same object

weather.param <- wth.param(weather, method="markov")

rain.compare <- rainfall(365*2, weather.param)
temp.compare <- temperature(365*2, weather.param)
```

```
data(soildat)
data(bufferdat)

# bluegrass buffer, clay loam soil
# short simulation to cut down on time required
vfs.CL <- VFS(nyears = 2, thissoil = subset(soildat, Soil == "CL"),
rain=rain.compare, temperature=temp.compare,
thisbuffer = subset(bufferdat, Species == "bluegrass"), Duration = 2,
FieldArea = 4000, VFSwidth = 10.7, VFSslope = 0.02,
z = 1000, b = 1.5)

print(vfs.CL)
summary(vfs.CL)

aple.CL <- VFSAPLE(vfs.CL, soilP = 120, OM = 2)

print(aple.CL)
summary(aple.CL)
```

---

## MUSLE — *Modified Universal Soil Loss Equation*

---

### Description

Simulation of soil erosion on a daily timestep.

### Usage

```
MUSLE(Q, qp, A, K, LS, C = 0.085, P = 0.40, a = 11.8, b = 0.56)
```

### Arguments

| | |
|---|---|
| Q | Runoff volume (m^3/d). |
| qp | Runoff peak discharge (m^3/s). |
| A | Field area (ha). |
| K | Soil erodibility factor. |
| LS | Landscape factor. |
| C | Crop management factor. Default is for a corn field. |
| P | Erosion control practice factor. |
| a | Location coefficient. Default value from Williams 1975. |
| b | Location coefficient. Default value from Williams 1975. |

## Details

Uses the Modified Universal Soil Loss Equation to estimate daily sediment yield. If K and LS are not known, they can be estimated from soil or field properties using `MUSLE.K` and `MUSLE.LS`.

If the location coefficients are known from measured sedimentation data, more accurate estimates can be made.

## Value

Sediment yield (t/day).

## Author(s)

Sarah Goslee

## References

Williams, J. R. (1975) Sediment-yield prediction with universal equation using runoff energy factor. Pp. 244–251 in: *Present and prospective technology for predicting sediment yield and sources.* ARS.S-40, US Gov. Print. Office, Washington, DC. 244-252.

Wischmeier, W. H. and Smith, D. D. (1978) *Predicting rainfall erosion losses-a guide to conservation planning.* U.S. Department of Agriculture, Agriculture Handbook No. 537.

## See Also

`MUSLE.K`, `MUSLE.LS`, `peak`

## Examples

```
# Approximate erodibility factor from soil texture.
Kf <- MUSLE.K(.3, .5, .2)

# Calculate landscape factor from field size and shape.
# 100-m field length with 2% slope
# note that MUSLE.LS takes feet
LS <- MUSLE.LS(100 * 3.28, .02)

# assume 0.4 ha cornfield with known rainfall intensity
peakd <- peak(intensity = 55, area = 0.4)

SedYield <- MUSLE(85, qp = peakd, A = .4, K = Kf, LS = LS)
```

MUSLE.K *Estimate soil erodibility factor K.*

## Description

Estimates MUSLE soil erodibility from a multiple regression model of soil texture.

## Usage

```
MUSLE.K(fc, fm, ff)
```

## Arguments

| | |
|---|---|
| fc | Fraction of coarse material (sand) in the soil (0-1). |
| fm | Fraction of medium material (silt) in the soil (0-1). |
| ff | Fraction of fine material (clay) in the soil (0-1). |

## Details

If K is not available from other sources, it can be estimated from soil texture (Goslee, in review).

## Value

Returns the soil erodibility factor K.

## Author(s)

Sarah Goslee

## References

Wischmeier, W. H. and Smith, D. D. (1978) *Predicting rainfall erosion losses-a guide to conservation planning.* U.S. Department of Agriculture, Agriculture Handbook No. 537.

## See Also

MUSLE

## Examples

```
# Approximate erodibility factor from soil texture.
Kf <- MUSLE.K(.3, .5, .2)

# Calculate landscape factor from field size and shape.
# 100-m field length with 2% slope
# note that MUSLE.LS takes feet
LS <- MUSLE.LS(100 * 3.28, .02)
```

```
# assume 0.4 ha cornfield with known rainfall intensity
peakd <- peak(intensity = 55, area = 0.4)

SedYield <- MUSLE(85, qp = peakd, A = .4, K = Kf, LS = LS)
```

---

MUSLE.LS *Estimate landscape factor LS*

---

## Description

Estimates MUSLE landscape factor for a homogeneous field from the field length and slope.

## Usage

```
MUSLE.LS(FieldLength, FieldSlope)
```

## Arguments

| | |
|---|---|
| FieldLength | Field length (ft). |
| FieldSlope | Field slope (ft/ft). |

## Value

MUSLE landscape factor LS.

## Author(s)

Sarah Goslee

## References

Wischmeier, W. H. and Smith, D. D. (1978) *Predicting rainfall erosion losses-a guide to conservation planning.* U.S. Department of Agriculture, Agriculture Handbook No. 537.

## See Also

MUSLE

## Examples

```
# Approximate erodibility factor from soil texture.
Kf <- MUSLE.K(.3, .5, .2)

# Calculate landscape factor from field size and shape.
# 100-m field length with 2% slope
# note that MUSLE.LS takes feet
LS <- MUSLE.LS(100 * 3.28, .02)
```

```
# assume 0.4 ha cornfield with known rainfall intensity
peakd <- peak(intensity = 55, area = 0.4)

SedYield <- MUSLE(85, qp = peakd, A = .4, K = Kf, LS = LS)
```

---

peak *Rational method to calculate peak discharge*

---

## Description

Very simple estimate of peak discharge.

## Usage

```
peak(intensity, area, c = 0.25)
```

## Arguments

| | |
|---|---|
| intensity | Precipitation intensity (mm/hr). |
| area | Field area (ha). |
| c | Runoff coefficient, related to slope, soil type, and land cover (0-1). Forest may be 0.05 - 0.25, while paved surfaces may be 0.95. |

## Value

Peak discharge (m^3/s).

## Author(s)

Sarah Goslee

## References

Hromadka, T. V, II and Whitley, R. J. (1994) The rational method for peak flow rate estimation. *Water Res Bull* **30**(6), 1001–1009.

## Examples

```
# peak discharge from a grassy meadow
peakd.meadow <- peak(intensity = 55, area = 1, c = 0.1)

# peak discharge from an urban area
peakd.urban <- peak(intensity = 55, area = 1, c = 0.8)
```

`print.APLE` *Printing the result of APLE*

## Description

Print method for APLE objects.

## Usage

```
## S3 method for class 'APLE'
print(x, ...)
```

## Arguments

| | |
|---|---|
| `x` | An APLE object produced by `APLE` or `VFSAPLE`. |
| `...` | Other arguments to print. |

## Details

Prints the annual mean for erosion, soil dissolved, manure dissolved, fertilizer dissolved, and total phosphorus losses.

## Value

Returns the APLE object x invisibly.

## Author(s)

Sarah Goslee

## See Also

`APLE`, `VFSAPLE`, `summary.APLE`,

## Examples

```
x <- APLE(soilP = 127, clay = 17, OM = 6, precip = 35, runoff = 6,
 erosion = 7, manureP = 25, manureSolids = 25, manureWEP = 50,
 manureIn = 40, fertP = 10, fertIn = 40)

print(x)
summary(x)
```

## print.VFS *Printing the result of VFS*

### Description

Print method for VFS objects.

### Usage

```
## S3 method for class 'VFS'
print(x, ...)
```

### Arguments

| | |
|---|---|
| x | A VFS x produced by VFS(). |
| ... | Other arguments to print. |

### Details

If the VFS object has a modeled filter strip, the mean annual loads in and out, as well as removal efficiencies are printed. Otherwise only the erosion values from the field are printed.

### Value

It returns the VFS object invisibly.

### Author(s)

Sarah Goslee

### See Also

VFS, summary.VFS

### Examples

```
# state college GHCN data
#
# weather <- read.dly(system.file("extdata", "USC00368449.dly", package = "VFS"))
data("weather") # same object

weather.param <- wth.param(weather, method="markov")

rain.compare <- rainfall(365*2, weather.param)
temp.compare <- temperature(365*2, weather.param)

data(soildat)
data(bufferdat)
```

```
# bluegrass buffer, clay loam soil
# short simulation to cut down on time required
vfs.CL <- VFS(nyears = 2, thissoil = subset(soildat, Soil == "CL"),
rain=rain.compare, temperature=temp.compare,
thisbuffer = subset(bufferdat, Species == "bluegrass"), Duration = 2,
FieldArea = 4000, VFSwidth = 10.7, VFSslope = 0.02,
z = 1000, b = 1.5)

print(vfs.CL)
summary(vfs.CL)
```

---

`rainfall` *Generate simulated daily rainfall*

---

## Description

Generates simulated daily rainfall based on parameters derived from daily weather data.

## Usage

```
rainfall(ndays, thiswth, months)
```

## Arguments

| | |
|---|---|
| `ndays` | Number of days to simulate. |
| `thiswth` | Output of `wth.param` The choice of method in `wth.param()` determines what coefficients are provided, and thus which rainfall simulation method is used. |
| `months` | If the rainfall simulation method uses monthly statistics (Markov), a vector of month numbers of `length(ndays)` may be provided. If it is missing, then January 1 is assumed to be the first day of a 365-day year. |

## Details

The rainfall simulation currently offers choice of two methods: the simple Poisson model of Rodriguez-Iturbe et al. (1999), and the Markov chain model of Nicks (1974). The latter rainfall calculation is used by the APEX farm model, among others, and is based on monthly statistics.

## Value

A vector of daily rainfall totals.

## Author(s)

Heather Gall and Sarah Goslee

### References

Rodriguez-Iturbe, I., Porporato, A., Ridolfi, L., Isham, V. and Coxi, D. R. (1999) Probabilistic modelling of water balance at a point: the role of climate, soil and vegetation. *Proc Royal Soc A* **455**, 269–288.

Nicks, A. D. (1974) Stochastic generation of the occurrence, pattern and location of maximum amount of daily rainfall. Pp. 154–171 in: *Proceedings Symposium on Statistical Hydrology.* USDA Agricultural Research Service Miscellaneous Publication No. 1275, Washington, DC.

### See Also

`wth.param`, `temperature`

### Examples

```
# GHCN daily weather file for State College, PA
# subset of data (2000-2009) for station USC00368449
#
data("weather") # same object


# calculate parameters for the poisson model
# using 0.3 mm as the lower limit for wet days.
weather.param.p <- wth.param(weather, method = "poisson", llim = 0.3)

# simulate ten years of rainfall
rain10.p <- rainfall(365*10, weather.param.p)

# increase per-event rainfall by 5 mm
weather.param.p5 <- weather.param.p
weather.param.p5$params$depth <- weather.param.p5$params$depth + 5
rain10.p5 <- rainfall(365*10, weather.param.p5)

# calculate parameters for the Markov chain model
# using 0.3 mm as the lower limit for wet days.
weather.param.m <- wth.param(weather, method = "markov", llim = 0.3)

# rainfall() selects Markov model based on input parameter types
rain10.m <- rainfall(365*10, weather.param.m)

# simulate 10 years of temperature
temp10 <- temperature(365*10, weather.param.p)
```

---

## read.dly *Read GHCN DLY daily weather file into a data frame*

---

### Description

Imports daily data files from the Global Historical Climatology Network (GHCN), replaces nodata values with NA, and converts precipitation to mm and temperature to C.

## Usage

```
read.dly(filename)
```

## Arguments

| | |
|---|---|
| `filename` | Filename or URL of a GHCN DLY file. |

## Details

All GHCN DLY files should have these five elements: PRCP (precipitation, originally tenths of a mm but mm in the function output); SNOW (snowfall, mm); SNWD (snow depth, mm), TMAX (maximum temperature, originally tenths of degree C but C in the function output), and TMIN (minimum temperature, originally tenths of degree C but C in the function output).

Depending on the station, there may be many other recorded variables. Each variable is accompanied by a series of quality flags, which are preserved from the original file.

Data are in a complex fixed-width format. Please see the GHCN readme for details.

## Value

Returns a data frame with date as three columns, YEAR, MONTH, DAY, and each data value present in the original file along with its quality flags. Please see the GHCN readme for details.

Note that units for temperature and precipitation have been converted from the GHCN values.

These columns will always be present in the output:

`YEAR` Year.

`MONTH` Month number.

`DAY` Day of month.

`PRCP` Precipitation (mm).

`TMAX` Maximum temperature (C).

`TMIN` Minimum temperature (C).

## Author(s)

Sarah Goslee

## References

GHCN data comprises both current and historical weather station data world-wide.

## See Also

`wth.param`, `weather`

### Examples

```
# ten years of daily weather data, 2000-2009, for State College, PA
weather <- read.dly(system.file("extdata", "USC00368449.dly", package = "VFS"))

# could also use:
# weather <- read.dly("ftp://ftp.ncdc.noaa.gov/pub/data/ghcn/daily/all/USC00368449.dly")
# weather <- subset(weather, YEAR >= 2000 & YEAR <= 2009)

# daily precipitation
summary(weather$PRCP.VALUE)

# monthly average maximum temperature
aggregate(TMAX.VALUE ~ MONTH, FUN = mean, data = weather)

# generate simulation values
weather.params <- wth.param(weather)
```

---

## soildat *Soil texture class properties*

---

### Description

Basic hydrologic properties for twelve soil texture classes.

### Usage

```
data("soildat")
```

### Format

A data frame with 12 observations on the following 9 variables.

`Soil` Texture class abbreviation.

`SoilName` Texture class name.

`Ksat` Saturated hydraulic conductivity (mm d^(-1)).

`ThetaSAT` Water potential at saturation.

`ThetaFC` Water potential at field capacity.

`ThetaWP` Water potential at wilt point.

`ff` Fraction of fine (clay) particles.

`fm` Fraction of medium (silt) particles.

`fc` Fraction of coarse (sand) particles.

### Source

Clapp, RB, Hornberger, GM. 1978. Empirical equations for some soil hydraulic properties. Water Resour Res 14:601-604. DOI: 10.1029/WR014i004p00601.

Karkanis, PG. 1983. Determining field capacity and wilting point using soil saturation by capillary rise. Can Agr Eng 25:19-21.

## Examples

```
# state college GHCN data
#
# weather <- read.dly(system.file("extdata", "USC00368449.dly", package = "VFS"))
data("weather") # same object

weather.param <- wth.param(weather, method="markov")

rain.compare <- rainfall(365*2, weather.param)
temp.compare <- temperature(365*2, weather.param)

data(soildat)
data(bufferdat)

# bluegrass buffer, clay loam soil
# short simulation to cut down on time required
vfs.CL <- VFS(nyears = 2, thissoil = subset(soildat, Soil == "CL"),
rain=rain.compare, temperature=temp.compare,
thisbuffer = subset(bufferdat, Species == "bluegrass"), Duration = 2,
FieldArea = 4000, VFSwidth = 10.7, VFSslope = 0.02,
z = 1000, b = 1.5)

print(vfs.CL)
summary(vfs.CL)

aple.CL <- VFSAPLE(vfs.CL, soilP = 120, OM = 2)

print(aple.CL)
summary(aple.CL)
```

---

summary.APLE *Summarize the result of APLE*

---

## Description

Summary method for APLE objects.

## Usage

```
 ## S3 method for class 'APLE'
summary(object, ...)
```

## Arguments

| | |
|---|---|
| object | APLE object produced by APLE. |
| ... | Other arguments to summary |

## Details

Calculates means for phosphorus loss.

## Value

Summary of APLE object.

## Author(s)

Sarah Goslee

## See Also

APLE, VFSAPLE, print.APLE,

## Examples

```
x <- APLE(soilP = 127, clay = 17, OM = 6, precip = 35, runoff = 6,
 erosion = 7, manureP = 25, manureSolids = 25, manureWEP = 50,
 manureIn = 40, fertP = 10, fertIn = 40)

print(x)
summary(x)
```

---

summary.VFS *Summarize the result of VFS*

---

## Description

Summary method for VFS objects.

## Usage

```
 ## S3 method for class 'VFS'
summary(object, ...)
```

## Arguments

| | |
|---|---|
| object | A VFS object produced by VFS(). |
| ... | Other arguments to summary |

## Details

Calculates means and standard deviations for annual removal efficiencies. If model type is erosion-only (no vegetated filter strip simulated), then only the relevant variables describing erosion from the field are returned.

### Value

| | |
|---|---|
| `ALR` | Annual load reduction. |
| `ALRsd` | SD of annual load reduction. |
| `ALR1000` | Load reduction across full timespan. |
| `ALR1000sd` | SD of load reduction across full timespan. |
| `APEA` | Annual per-event average reduction. |
| `APEAsd` | SD of annual per-event average reduction. |
| `SedIn` | Sediment entering the vegetated filter strip per year. |
| `SedInsd` | SD of sediment entering the vegetated filter strip per year. |
| `SedLoss` | Sediment lost per year. |
| `SedLosssd` | SD of sediment lost per year. |
| `TLR` | Total load reduction. |
| `RunoffDays` | Days per year with runoff. |
| `RunoffDayssd` | SD of days per year with runoff. |
| `Days` | Total number of days. |

### Author(s)

Sarah Goslee

### See Also

`VFS`, `print.VFS`

### Examples

```
# state college GHCN data
#
# weather <- read.dly(system.file("extdata", "USC00368449.dly", package = "VFS"))
data("weather") # same object

weather.param <- wth.param(weather, method="markov")

rain.compare <- rainfall(365*2, weather.param)
temp.compare <- temperature(365*2, weather.param)

data(soildat)
data(bufferdat)

# bluegrass buffer, clay loam soil
# short simulation to cut down on time required
vfs.CL <- VFS(nyears = 2, thissoil = subset(soildat, Soil == "CL"),
rain=rain.compare, temperature=temp.compare,
thisbuffer = subset(bufferdat, Species == "bluegrass"), Duration = 2,
FieldArea = 4000, VFSwidth = 10.7, VFSslope = 0.02,
z = 1000, b = 1.5)
```

```
print(vfs.CL)
summary(vfs.CL)
```

---

`temperature` *Generate simulated mean temperature*

---

## Description

Generates simulated daily temperature minimum and maximum based on parameters derived from daily weather data.

## Usage

```
temperature(ndays, thiswth)
```

## Arguments

| | |
|---|---|
| `ndays` | number of days to simulate |
| `thiswth` | list output of `wth.param`. |

## Details

This is a very simple temperature simulation, using three parameters derived from daily weather data and the day of year to calculate a smooth annual temperature change derived from the first harmonic of a Fourier function.

## Value

Returns a vector of daily mean temperature (X).

## Author(s)

Heather Gall and Sarah Goslee

## References

Grimenes, A. and Nissen, O. (2004) Mathematical modeling of the annual temperature wave based on monthly mean temperatures, and comparisons between local climate trends at seven Norwegian stations. *Theor Appl Climatol* **78**, 229–246. doi:10.1007/s0070400400369

## See Also

`wth.param`, `rainfall`

## Examples

```
# A sample GHCN daily weather file for State College, PA, is included with this package.
# This file contains a subset of data (1980-2009) for station USC00368449
data("weather")

# calculate parameters for the poisson model, using 0.3 mm as the lower limit for wet days.
weather.param.p <- wth.param(weather, method = "poisson", llim = 0.3)

# simulate 10 years of temperature
temp10 <- temperature(365*10, weather.param.p)
```

---

`USC00368449.dly` *GHCN Data for State College, PA, 1980-2009*

---

## Description

A ten-year sample of Global Historical Climatology Network (GHCN) daily weather data for State College, PA (station USC00368449). This is in a complex fixed-width format. Please see the GHCN readme for details.

## Usage

```
"USC00368449.dly"
```

## Format

A complex fixed-width ASCII file.

## Source

GHCN data comprises both current and historical weather station data world-wide.

## See Also

`read.dly`

## Examples

```
# A sample DLY file for State College, PA, is included with this package.
# This file contains a subset of data (1980-2009) for station USC00368449
#
weather <- read.dly(system.file("extdata", "USC00368449.dly", package = "VFS"))
# identical to data("weather")
```

VFS *Vegetated filter strip and erosion model*

## Description

Simulated erosion and runoff given climate and soil texture, with or without a vegetated filter strip in place.

## Usage

```
VFS(nyears = 1000, thissoil, thisbuffer, rain, temperature,
  Duration = 2, FieldArea = 4000, VFSwidth = 10.7, VFSslope = 0.02,
  FieldSlope,  z = 1000, a = 1, b = 1.5,
  carrysoilwater = TRUE, runoffcalc = TRUE)
```

## Arguments

| | |
|---|---|
| nyears | Number of years to simulate. |
| thissoil | Soil properties for the site, as from soildat. |
| thisbuffer | Vegetation properties for the buffer strip, as from bufferdat. |
| rain | Daily rainfall (mm). |
| temperature | Daily mean temperature (C). |
| Duration | Rainfall event length. Default is 2 hours. |
| FieldArea | Field area (m^2). |
| VFSwidth | Filter strip width (m). |
| VFSslope | Filter strip slope (m/m). |
| FieldSlope | Optional field slope (m/m). If missing, VFSslope will be used. |
| z | Rooting zone depth (mm). Default is 1000 mm. |
| a | Empirical parameter that relates concentration and flow in the concentration-discharge relationship, C = aQ^b. |
| b | Empirical parameter that relates concentration and flow in the concentration-discharge relationship, C = aQ^b. May be a single value or a vector of values. |
| carrysoilwater | Boolean describing whether to store soil water; if FALSE, soil is always at field capacity. This option allows the effect of soil water storage to be quantified. |
| runoffcalc | Boolean describing whether to use intensity and saturation exceedances; if FALSE, all rainfall becomes runoff. This option allows the effect of runoff calculation to be quantified. |

## Details

The concentration-discharge (C-Q) model of erosion is intended to produce relative erosion values, rather than absolute values, but will produce absolute values if a and b are known. The MUSLE field erosion model is run alongside the C-Q model. The K factor is estimated from soil texture data using `MUSLE.K`, and the LS factor from field properties using `MUSLE.LS`. Blaney-Criddle coefficients for evapotranspiration calculations from a cornfield are hard-coded; a future update will allow for varying the type of field.

## Value

Returns an object of class VFS, comprising:

`daily` Daily output of all public variables that change as a function of time. The data frame has columns:

- rain: precipitation (mm).
- temperature: mean temperature (C).
- S: soil water storage, (mm).
- kt: Blaney-Criddle temperature coefficient.
- ET: evapotranspiration (mm).
- intensity: rainfall intensity (mm).
- runoff: runoff (mm).
- Q: discharge (ft^3/s).
- fd: flow depth through VFS (ft).
- R: hydraulic radius of filter strip (ft).
- Vm: Manning's velocity (ft/s).
- Re: Reynold's number.
- Va: actual shear stress (ft/s).
- Nfc: Fall number for coarse particles.
- Nfm: Fall number for medium particles.
- Nff: Fall number for fine particles.
- fdc: Trapping efficiency for coarse particles.
- fdm: Trapping efficiency for medium particles.
- fdf: Trapping efficiency for fine particles.
- Ft: Total trapping efficiency of filter strip.
- peakflow: peak flow (m^3/s).

`field` Data on the field being modeled:

- clay: soil clay content (%.)
- area: field area (m^2).

| | |
|---|---|
| `Conc` | Sediment concentration (in mass/volume) as calculated by the relationship C = aQ^b; specific units depend on units conversions included in the value of a. |
| `MassIn` | Sediment load (mass) from the C-Q model, as calculated by multiplying concentration and runoff volume. If concentration is assumed to be in g/L, then the load is calculated in g. |
| `MassOut` | Sediment mass from the C-Q model that leaves the vegetated filter strip at the end of a runoff event (i.e., the mass that is not removed). |
| `MassRemoved` | Sediment mass from the C-Q model that remains in the vegetated filter strip at the end of a runoff event. |
| `AnnualMassIn` | Sum of the sediment loads from the C-Q model entering the vegetated filter strip over the course of one year. |
| `AnnualMassOut` | Sum of the sediment loads from the C-Q model leaving the vegetated filter strip over the course of one year. |
| `AnnualRemovalEfficiency` | The removal effficiency from the C-Q model of the vegetated filter strip at an annual time scale (%). |
| `MassInMUSLE` | Sediment mass from the MUSLE model leaving the crop field . |
| `MassOutMUSLE` | Sediment mass from the MUSLE model that leaves the vegetated filter strip at the end of a runoff event (t/day). |
| `MassRemovedMUSLE` | Sediment mass that remains in the vegetated filter strip at the end of a runoff event (t/day). |
| `AnnualMassInMUSLE` | Sum of the sediment loads entering the vegetated filter strip over the course of one year (t). |
| `AnnualMassOutMUSLE` | Sum of the sediment loads leaving the vegetated filter strip over the course of one year (t). |
| `AnnualRemovalEfficiencyMUSLE` | The removal effficiency of the vegetated filter strip at an annual time scale (%). |
| `Ftannual` | Filter strip removal efficiency. |
| `Ftannualavg` | The average of all per-event trapping efficiencies over the course of one year. |

## Author(s)

Heather Gall, Sarah Goslee, and Tamie Veith

## References

Gall, H. E., Schultz, D., Veith, T. L, Goslee, S. C., Mejia, A., Harman, C. J., Raj, C. and Patterson, P. H. (2018) The effects of disproportional load contributions on quantifying vegetated filter strip sediment trapping efficiencies. *Stoch Environ Res Risk Assess* **32**(8), 2369–2380. doi:10.1007/s004770171505x

Haan C. T., Barfield B. J. and Hayes J. C. (1994) *Design hydrology and sedimentology for small catchments.* Acad Press, San Diego.

Williams, J. R. (1975) Sediment-yield prediction with universal equation using runoff energy factor. Pp. 244–251 in: *Present and prospective technology for predicting sediment yield and sources.* ARS.S-40, US Gov. Print. Office, Washington, DC. 244-252.

Wischmeier, W. H. and Smith, D. D. (1978) *Predicting rainfall erosion losses-a guide to conservation planning.* U.S. Department of Agriculture, Agriculture Handbook No. 537.

## See Also

print.VFS, summary.VFS, wth.param, soildat, bufferdat,

## Examples

```
# state college GHCN data
#
# weather <- read.dly(system.file("extdata", "USC00368449.dly", package = "VFS"))
data("weather") # same object

weather.param <- wth.param(weather, method="markov")

rain.compare <- rainfall(365*2, weather.param)
temp.compare <- temperature(365*2, weather.param)

data(soildat)
data(bufferdat)

# bluegrass buffer, clay loam soil
# short simulation to cut down on time required
vfs.CL <- VFS(nyears = 2, thissoil = subset(soildat, Soil == "CL"),
rain=rain.compare, temperature=temp.compare,
thisbuffer = subset(bufferdat, Species == "bluegrass"), Duration = 2,
FieldArea = 4000, VFSwidth = 10.7, VFSslope = 0.02,
z = 1000, b = 1.5)

print(vfs.CL)
summary(vfs.CL)

aple.CL <- VFSAPLE(vfs.CL, soilP = 120, OM = 2)

print(aple.CL)
summary(aple.CL)
```

---

# VFSAPLE *Link the VFS and APLE models.*

---

## Description

Uses the erosion and runoff output of VFS as input to APLE.

## Usage

```
VFSAPLE(x, soilP, OM, manureP = 25, manureSolids = 25, manureWEP = 50,
    manureIn = 40, fertP = 10, fertIn = 40)
```

## Arguments

| | |
|---|---|
| `x` | VFS object. |
| `soilP` | soil test Mehlich 3 phosphorus (mg/kg). |
| `OM` | soil organic matter (%). |
| `manureP` | manure P applied (kg/ha). |
| `manureSolids` | manure solids (%). |
| `manureWEP` | manure water-extractable phosphorus/TP (%). |
| `manureIn` | manure incorporated (%). |
| `fertP` | fertilizer phosphorus applied (kg/ha). |
| `fertIn` | fertilizer incorporated (%). |

## Details

The `VFSAPLE` function handles unit conversion and parameter estimation. Erosion, precipitation, runoff, and field parameters from `VFS` are passed to `APLE`. It also runs `APLE` for both pre- and post-filter strip erosion values, so the efficacy of the filter strip at phosphorus removal can be calculated.

## Value

| | |
|---|---|
| `preVFS` | APLE object for pre-filter strip erosion values. |
| `postVFS` | APLE object for post-filter strip erosion values. |
| `pErosion` | Efficacy of the filter strip at removing erosion phosphorus (%). |
| `pTotal` | Efficacy of the fiter strip at removing total phosphorus (%). |

## Author(s)

Sarah Goslee

## See Also

`APLE`, `VFS`

## Examples

```
# state college GHCN data
#
# weather <- read.dly(system.file("extdata", "USC00368449.dly", package = "VFS"))
data("weather") # same object

weather.param <- wth.param(weather, method="markov")
```

```
rain.compare <- rainfall(365*2, weather.param)
temp.compare <- temperature(365*2, weather.param)

data(soildat)
data(bufferdat)

# bluegrass buffer, clay loam soil
# short simulation to cut down on time required
vfs.CL <- VFS(nyears = 2, thissoil = subset(soildat, Soil == "CL"),
rain=rain.compare, temperature=temp.compare,
thisbuffer = subset(bufferdat, Species == "bluegrass"), Duration = 2,
FieldArea = 4000, VFSwidth = 10.7, VFSslope = 0.02,
z = 1000, b = 1.5)

print(vfs.CL)
summary(vfs.CL)

aple.CL <- VFSAPLE(vfs.CL, soilP = 120, OM = 2)

print(aple.CL)
summary(aple.CL)
```

---

## weather *Ten years of daily weather data*

---

### Description

The VFS offers the capability of importing weather data from the GHCN, either from local files or the online repository, but this import is slow, so the result of the import is saved as an R object for those examples that need it.

The original GHCN file has many more columns.

### Usage

```
data("weather")
```

### Format

A data frame with 3653 observations on the following 8 variables.

YEAR a numeric vector

MONTH a numeric vector

DAY a numeric vector

PRCP.VALUE a numeric vector

SNOW.VALUE a numeric vector

SNWD.VALUE a numeric vector

TMAX.VALUE a numeric vector

TMIN.VALUE a numeric vector

### Source

GHCN data comprises both current and historical weather station data world-wide.

### See Also

wth.param, read.dly

### Examples

```
# state college GHCN data
#
# created by:
# weather <- read.dly(system.file("extdata", "USC00368449.dly", package = "VFS"))
data("weather") # same object: 10 years of daily weather data

weather.param <- wth.param(weather, method="markov")

rain.compare <- rainfall(365*3, weather.param)
temp.compare <- temperature(365*3, weather.param)
```

---

## wth.param — *Calculate weather parameters from daily data for use in climate simulations*

---

### Description

The climate generation functions for rainfall and temperature require parameters calculated from GHCN daily weather data, or from any data frame with columns containing year, month, day, precipitation, and minimum and maximum temperature. Partial years at the beginning or end of the dataset are removed. Leap days are also removed to standardize day-of-year calculation.

### Usage

```
wth.param(dly, llim = 0, method = "poisson", year.col = "YEAR",
 month.col = "MONTH", day.col = "DAY", prcp.col = "PRCP.VALUE",
 tmin.col = "TMIN.VALUE", tmax.col = "TMAX.VALUE")
```

### Arguments

| | |
|---|---|
| dly | A data frame, such as the output of read.dly(), with days as rows and columns including YEAR, MONTH, DAY, PRCP.VALUE, TMIN.VALUE, TMAX.VALUE (for GHCN data), or with those columns having names specivied by arguments. |
| llim | The minimum daily rainfall for a wet day to be counted. |
| method | Choice of model for which to calculate parameters, either "poisson" or "markov". |
| year.col | Name of the column containing year number. |
| month.col | Name of the column containing month number. |

| | |
|---|---|
| `day.col` | Name of the column containing day number. |
| `prcp.col` | Name of the column containing daily precipitation. |
| `tmin.col` | Name of the column containing daily minimum temperature. |
| `tmax.col` | Name of the column containing daily maximum temperature. |

## Details

The rainfall simulation currently offers choice of two methods: the simple Poisson model of Rodriguez-Iturbe et al. (1999), and the Markov chain model of Nicks (1974). The latter rainfall calculation is used by the APEX farm model, among others, and is based on monthly statistics. NOTE: For reasons of time and space, the example contains only ten years of daily weather data. We suggest using thirty years for estimating parameter values.

## Value

`params` Parameters for simulating long-term point rainfall. For `method = "poisson"`, a list of parameters containing:

- lambda: Mean rainfall inter-arrival frequency (d-1).
- depth: Mean rainfall depth (mm).
- A: Mean annual temperature (C).
- B: Temperature half-amplitude (C).
- C: Day of the year with minimum temperature (DOY).
- start: First full year of the weather data.
- end: Last full year of the weather data.

For `method = "markov"`, a data frame with one row per month and columns:

- tmin: Minimum temperature.
- tminsd: Maximum temperature.
- tmax: Minimum temperature standard deviation.
- tmaxsd: Maximum temperature standard deviation.
- prcp: Monthly precipitation.
- prcpmean: Mean size of a precipitation event.
- prcpmax: Maximum size of a precipitation event.
- prcpsd: Standard deviation of precipitation event sizes.
- prcpskew: Skew of precipitation event sizes.
- prcpwet: Number of wet days (greater than `llim`).
- prcpww: Probability of a wet day following a wet day.
- prcpdw: Probability of a wet day following a dry day.

| | |
|---|---|
| `temperature` | Parameters for simulating long-term daily temperature. |
| `llim` | Minimum daily rainfall for a wet day. |
| `start` | First full year of weather data |
| `end` | Last full year of weather data |

### Author(s)

Sarah Goslee

### References

Rodriguez-Iturbe, I., Porporato, A., Ridolfi, L., Isham, V. and Coxi, D. R. (1999) Probabilistic modelling of water balance at a point: the role of climate, soil and vegetation. *Proc Royal Soc A* **455**, 269–288.

Nicks, A. D. (1974) Stochastic generation of the occurrence, pattern and location of maximum amount of daily rainfall. Pp. 154–171 in: *Proceedings Symposium on Statistical Hydrology.* USDA Agricultural Research Service Miscellaneous Publication No. 1275, Washington, DC.

### See Also

read.dly, rainfall, temperature

### Examples

```
# GHCN daily weather file for State College, PA
# subset of data (2000-2009) for station USC00368449
#
data("weather")

# calculate parameters for the poisson model
# using 0.3 mm as the lower limit for wet days.
weather.param.p <- wth.param(weather, method = "poisson", llim = 0.3)

# simulate ten years of rainfall
rain10.p <- rainfall(365*10, weather.param.p)

# increase per-event rainfall by 5 mm
weather.param.p5 <- weather.param.p
weather.param.p5$params$depth <- weather.param.p5$params$depth + 5
rain10.p5 <- rainfall(365*10, weather.param.p5)

# calculate parameters for the Markov chain model
# using 0.3 mm as the lower limit for wet days.
weather.param.m <- wth.param(weather, method = "markov", llim = 0.3)

# rainfall() selects Markov model based on input parameter types
rain10.m <- rainfall(365*10, weather.param.m)

# simulate 10 years of temperature
temp10 <- temperature(365*10, weather.param.p)
```

# Index


∗ **IO**
read.dly, 14
∗ **datagen**
rainfall, 13
temperature, 20
∗ **datasets**
bufferdat, 5
soildat, 16
USC00368449.dly, 21
weather, 27
∗ **methods**
summary.APLE, 17
summary.VFS, 18
∗ **package**
VFS-package, 2
∗ **print**
print.APLE, 11
print.VFS, 12

APLE, 3, *11*, *18*, *26*

bufferdat, 5, *25*

MUSLE, 6, *8*, *9*
MUSLE.K, *7*, 8
MUSLE.LS, *7*, 9

peak, *7*, 10
print.APLE, 11, *18*
print.VFS, 12, *19*, *25*

rainfall, 13, *20*, *30*
read.dly, 14, *21*, *28*, *30*

soildat, 16, *25*
summary.APLE, *11*, 17
summary.VFS, *12*, 18, *25*

temperature, *14*, 20, *30*

USC00368449.dly, 21

VFS, *12*, *19*, 22, *26*
VFS-package, 2
VFSAPLE, *4*, *11*, *18*, 25

weather, *15*, 27
wth.param, *14*, *15*, *20*, *25*, *28*, 28